

Karkar A, Mak T, Dahir N, Al-Dujaily R, Tong KF, Yakovlev A.
[Network-on-Chip Multicast Architectures Using Hybrid Wire and](#)
[Surface-Wave Interconnects.](#)

***IEEE Transactions on Emerging Topics in Computing* 2016**

DOI: <http://dx.doi.org/10.1109/TETC.2016.2551043>

Copyright:

© 2016 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.

DOI link to article:

<http://dx.doi.org/10.1109/TETC.2016.2551043>

Date deposited:

06/04/2016

Network-on-Chip Multicast Architectures Using Hybrid Wire and Surface-Wave Interconnects

Ammar Karkar¹, Terrence Mak², Nizar Dahir³, Ra'ed Al-Dujaily¹, Kin-Fai Tong⁴, and Alex Yakovlev¹

¹School of Electrical and Electronic Engineering, Newcastle University, UK, Email: {a.j.m.karkar, raaed.aldujaily, alex.yakovlev}@newcastle.ac.uk

²Faculty of Physical Sciences and Engineering, University of Southampton, Southampton, United Kingdom, Email: tmak@ecs.soton.ac.uk

³Department of Electronics, University of York, York, UK, Email: nizar.dahir@york.ac.uk

⁴Department of Electrical and Electronic Engineering, UCL, London, UK, Email: K.tong@ucl.ac.uk

Abstract—The network-on-chip (NoC) has been introduced as an efficient communication backbone to tackle the increasing challenges of on-chip communication. Nevertheless, merely metal-based NoC implementation offers only limited performance and power scalability in terms of multicast and broadcast traffics. To meet scalability demands, this paper addresses the system-level challenges for intra-chip multicast communication in a proposed hybrid interconnects architecture. This hybrid NoC combines and utilizes both regular metal on-chip interconnects and new type of wireless-NoC (WiNoC) which is Zenneck surface wave interconnects (SWI). Moreover, this paper embeds novel multicast routing and arbitration schemes to address system-level multicast-challenges in the proposed architecture. Specifically, a design exploration of contention handling in SWI layer is considered in both centralized and decentralized manners. Consequently, the hybrid wire-SWI architecture avoids overloading the network, alleviates the formation of traffic hotspots and avoid deadlocks that are typically associated with state-of-the-art multicast handling. The evaluation is based on a cycle-accurate simulation and hardware description. It demonstrates the effectiveness of the proposed architecture in terms of power consumption (up to around 10x) and performance (around 22x) compared to regular NoCs. These results are achieved with negligible hardware overheads. This study explore promising potential of the proposed architecture for current and future NoC-based many-core processors.

Keywords—*Networks-on-chip, surface wave, chip multiprocessors, on-chip interconnects, multicast routing, arbitration.*

I. INTRODUCTION

Multi/many-core processors had been introduced to provide near linear performance improvements as complexity increases (overcoming Pollack's rule) while maintaining lower power and frequency budgets [1]. Consequently, many-core era with hundreds of cores is upon us. However, a good utilization of such many-core architectures is becoming a challenge since the performance and power consumption of many-cores are bound by both interconnect fabric and cache coherence protocols. Although, networks-on-chip (NoC) have been adopted as a scalable underlying one-to-one communication structure [2], cache coherence protocols inject a non-trivial percentages of multicast/broadcast packets. This one-to-many (1-to-M) traffic is ranging from 3% to 13% [3].

In the literature, NoC conventionally treat 1-to-M traffic patterns as repeated unicast traffic, which is referred to as soft-

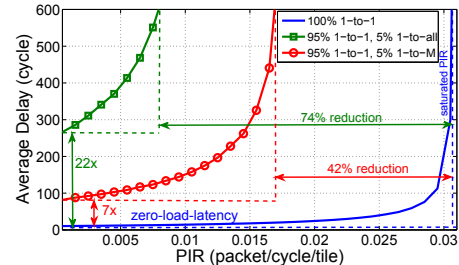


Fig. 1: Our 6 × 6 regular mesh NoC simulations with random traffic with a small percentage of multicast or broadcast (5%). The introduction of multicast or broadcast leads to severe deterioration in performance in terms of latency and saturation PIR.

ware multicast. This basic handling of multicast will increase NoC power consumption and congestion. Consequently, even small ratios of multicast (1-to-M) or broadcast (1-to-all) will have severe effects on NoC such as high latency and fast NoC saturation, see Fig. 1. Many studies have proposed wire-based NoC schemes that support 1-to-M communication[3], [4]. However, these studies struggles to match wire-latency and/or wire-energy. This will not be sufficient given the wire issues in terms of latency and energy for even unicast communication [5], [6].

As a result, many researchers are looking for alternative communication fabrics, such as radio frequency(RF)-based interconnects [7], [8], [9], [10] and optical interconnects (ONoC) [11], [12]. However, although such interconnects seem promising, they might not be the ideal solution due to their complexity, incompatibility, power consumption and/or area overheads [5]. The Zenneck surface wave (SW) [13], [14] is an emerging wireless on-chip interconnect technology which is exploited in this paper to mitigate global 1-to-M communication issues. The remarkable potential of SW requires innovative designs at different levels of abstraction in order to be utilized for future on-chip interconnects. This paper investigates the potential merits of the SW in handling 1-to-M on-chip communication and the associated challenges at the network abstraction level. A preliminary version of this work has been published previously [15], [16]. This paper offers new revised and improved designs in terms of multicast handling along with comparative evaluation of the proposed approaches. The major contributions of this paper are:

- To develop a hybrid wire-surface wave interconnect

(W-SWI) architecture that exploits surface wave (SW) features for 1-to-M traffic handling. Moreover, the SW features and challenges are analysed and discussed compared to emerging state-of-the-art interconnects.

- To propose arbitration mechanisms, routing scheme and communication protocols for 1-to-M traffic that efficiently address multicast traffic and maximize W-SWI utilization. In particular, in a design exploration of the centralized and decentralized arbitration techniques, they are demonstrated to have the ability to allow the concurrent utilization of many resources with relatively low circuit complexity and delay.
- To evaluate rigorously the W-SWI for both synthetic traffic and real application benchmarks. The proposed architectures are found to surpass the previous work, such as regular mesh by achieving improvements ($\sim 22\times$) in average delay and ($\sim 2 - 10\times$) in power consumption with relatively insignificant additional hardware cost.

The rest of the paper is organized as follows: Section II provides an overview of fan-out capability of emerging on-chip interconnects. Section III presents a hybrid wire-SW interconnect architecture, a multicast routing scheme, a design exploration of SWI arbitration techniques. Section IV evaluates the performance, power consumption and area overhead of the platform. Finally, Section V draws the conclusions and suggests directions for future work.

II. BACKGROUND

A. Fan-out Challenges and Emerging Interconnects

Power efficiency decreases significantly proportional to the number of fan-out in cutting emerging on-chip interconnects. Firstly, optical-based multicast architectures vary in topology and the on-chip devices that support them. For example, the tree-topology requires splitters and combiners to fork and join the optical signals [17]. Another example is a bus-based topology that utilizes wavelength-division-multiplexing (WDM) and then uses a bank of microring modulators, which can be configured to listen to a selected channel [11]. However, all these architectures have limited fan-out capability because the optical signal would decay after each forking or partial drop of the signal to a receiver node [11], [17]. The number of nodes that can receive the signal depend on the signal power budget, which is considered to be relatively high.

Secondly, RF-based interconnects (WiNoC, RF-I, and SWI) appears to be a cost effective alternative compared to optical interconnects since the RF circuitry require less complex implementation techniques and are less area and power-hungry. In terms of RF-I multicast architectures, many designs have proposed a worm or cycle layout of transmission lines (TLs) to pass through all the nodes. This layout involves a set of challenges such as adding nontrivial area overheads, signal decay and signal latency. In terms of fan-out feature, to distribute the signal in RF-I, the worm or cycle layout of these thick wires should go through almost every tile in the chip [8], [9]. Although this add nontrivial area overheads because of the large pitch of the TLs (width and spacing), the main issue is the multicast scalability. This is due to fact that this layout might

mitigate but not eliminate the impedance discontinuity. As a result, with each drop point, the signal are decayed, latency and signal reflections are increased unless careful matching circuits are designed [18].

On the other hand, The WiNoC have natural scalable fan-out capability which makes them preferable for 1-to-M enabled interconnect architectures. As a result, WiNoC had been suggested for multi/many-core with multicast requirements [19]. However, the WiNoC fan-out capability is limited by the antenna radiation pattern and coverage distance. The high power dissipation of the RF signal in the free space propagation leads to a low coverage distance to power ratio. Therefore, the transceiver power amplifier and the antenna design should take into consideration the required distance and the directions of the destinations. For instance, some studies have proposed run-time tunable transmitting power based on the required destination [20]. SWI is new type of WiNoC that will be discussed in next section.

B. Zenneck Surface Wave Interconnect (SWI)

The Zenneck surface wave is an inhomogeneous 2D electromagnetic (EM) wave supported by a surface. The designed surface is a waveguide that traps the EM in two dimensional media instead of three dimensional free space. As a result, the electrical-field decay rate in the SW from the source horizontally along the boundary is around $(1/\sqrt{d})$ as shown in Fig. 2, where d is the distance from the source [14]. Thus, since the SW signal transmitted in all directions, the SWI interconnect offers natural efficient fanout features compared to other interconnects [21]. In addition, This low power dissipation allows the SWI to offer relatively linear J/bit over this short distance compared to the high scaling of regular global buffered wire interconnects. The surface should be engineered by altering its dimensions, and the materials of the conductor and/or dielectric are chosen so that the characteristic impedance (Z_0) will be around $(10 + j300) \Omega$. Thus, the surface medium can consist of either a dielectric coated conductor layer or a corrugated conductor surface [13], [14]. For low fabrication costs and simple geometry, the dielectric-coated surface is preferable. The integrated surface can be realised using either silicon dioxide ($SiO_2, \epsilon_r = 3.9$) or ceramic ($Al_2O_3, \epsilon_r = 9.8$), on a metal ground plane of thickness $1\mu m$. In the case of millimetre-wave applications at 60 GHz or above, the thickness of the dielectric layer for silicon dioxide and ceramic will be $0.8mm$ and $0.7mm$ respectively, and the coating process can be integrated with a conventional semiconductor fabrication process. In addition, due to the fact that the surface roughness of the dielectric layer will not be an issue when the operating frequency of the system is less than 300 GHz, no expensive highly polished wafer ($Ra < 0.01\mu m$). As a result, the cost of the additional process can be neglected.

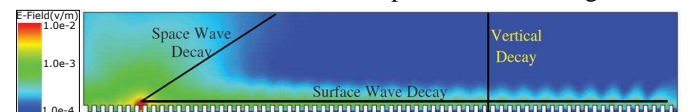


Fig. 2: Zenneck surface wave signal decay, which is significantly better than wireless free space signal decay [14].

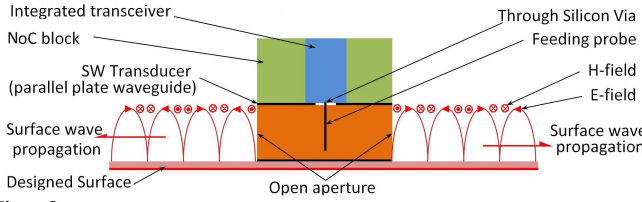


Fig. 3: Integrated transceiver and integrated transducer (inverted quarter-wavelength monopole) stacked over the designed surface.

Laboratory experiments [13] show that frequency bandwidth is limited only by the transceiver. However, integrated transceiver carrier frequencies are continuously scaling with the switching speed of the CMOS technology. This range of frequencies is necessary to allow multi-channel realization based on frequency-division-multiple-access (FDMA) at this shared media with the necessary frequency spacing to avoid channel interference. Thus, an integrated transceiver can be designed for a waveguided signal such as the one proposed and implemented by Chang *et al.* [22] or Carpenter *et al.* [9]. The communication channel is designed so that each channel has 32 sub-channels where each sub-channel transmits a nibble (4 bits) after it has been modulated using 16-QAM (quadrature amplitude modulation). This way the channel matches the data bandwidth of the baseline architecture wire link, see Section IV. Details of communication channel specifications are discussed in detail in previous work [23].

In addition, a maximum transmission into the Zenneck surface wave occurs when the incoming wave is incident at or close to the Brewster angle, where reflections are minimized. Therefore, an integration of a transducer linked to the transceiver is needed to launch the waved signal into the surface [13], as shown in Fig. 3. This can be as simple as, for omni-directional transmission, a coaxial to waveguide flange as described elsewhere [14]. Also, it could be a dipole or monopole for omni-directional communication, with a parallel plate waveguide [24]. The transducer layer can be fabricated separately and then flip-chip bonding and the through-silicon-via (TSV) [25] technique are used to connect it to the integrated transceiver. The transducer and transceiver design is beyond the scope of this paper.

III. PROPOSED SURFACE-WAVE-BASED MULTICAST ARCHITECTURES

A. Hybrid Wire and Surface-Wave Interconnect Architecture

Unlike application specific SoC interconnects [26], many-cores, where a range of applications are running, sometimes simultaneously, require general purpose interconnects with full connectivity yet with ability to satisfy different traffic patterns. Hybrid NoCs could retain the 1-to-M capability of the buses and reduce inter-node average hop count while maintaining high interconnect scalability if a high performance interconnect was adopted as the bus network layer.

The SWI has significant advantages especially in terms of fan-out, as mentioned earlier. However, as with all RF-based interconnects, it suffers from limitations in terms of congested shared media and limited range of frequencies. These make it infeasible to completely replace metal wire interconnects in the

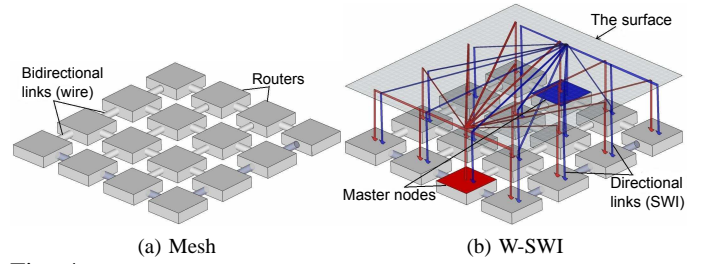


Fig. 4: Example showing that inserting two SWI channels in the proposed hybrid wire-SWI multilayer-network increases the overall NoC bisection bandwidth: (a) conventional on-chip network layer with 4-ary 2-mesh topology; (b) connections of both layers, metal wire and SWI.

near future. Moreover, in terms of wire-based interconnects, local communication seems to scale well with technology scaling unlike global communication [5]. In addition, this type of interconnect has the cheapest implementation cost compared to other fabrics. Therefore, the best solution would be to combine both metal and SWI in hybrid wire-SW interconnects in a multi-layered network architecture; in short W-SWI, as shown in Fig. 4. The first layer is a regular mesh topology, since the mesh is preferable for a general purpose interconnects architecture, suitable for chip floor planning, and have uniform manageable lengths of wires. On the other hand, the second layer is the surface wave bus topology. Thus, this architecture offers a natural fan-out feature, which is lost when the interconnects system changes from the bus to the NoC.

In order to preserve the fan-out feature, all the routers in the NoC are designed to receive information through SWI. On the other hand, even though enabling all nodes to have transmission capability will increase connectivity, this would increase the contention on the SWI layer as the NoC size increases. In addition, multicast communication are relatively low but with a dramatic effect on NoCs performance, as mentioned earlier. Therefore, fewer nodes are selected to have the transmission capability to reduce the circuit overhead and comply with the available frequency bandwidth. These nodes will be referred to as masters, while the rest are referred to as slaves. These slaves can only receive data but they may transmit some control signals, see section III-E. Masters are distributed so that the average hop count (Manhattan distance) from all slaves to the nearest master is at a minimum. Such placement of the master nodes reduces the average hop count of the overall on-chip network. In addition, it would allow each master to be accessible with minimum number of hops via wires and routers for critical traffic such as the 1-to-M.

The wire-based NoC mesh topology with software-multicast is the baseline architecture, and we will refer to it as the MeshS. For the W-SWI architecture proposed in this study, a sixth port needs to be added to the router along with all related control circuits. Also the crossbar switch size needs to be adjusted according to the new requirements. The new data path port is linked to a transceiver for master nodes or receiver for slave nodes.

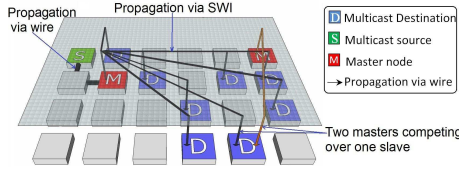


Fig. 5: W-SWI improved tree-based multicast routing with low latency packet delivery where branching possible only at the SWI master nodes.

B. Hybrid Multicast Routing Scheme

Multicast traffic benefits from the characteristics of the SWI and the architectural layout, as mentioned in previous sections. However, a smart routing technique is required to direct and deliver multicast traffic to its final destinations and to maximise the benefits at minimum cost. In this work, the routing for the proposed architecture is an improved tree-based scheme where the embedded tree path forks at one point; specifically, the nearest master. Therefore, the maximum degree of routing graph is up to $(N - 1)$; thus, the need for the SW fan-out feature. The nearest master then delivers concurrently the flit to all the addressed leaves (slaves) in one hop via SWI as illustrated in Fig. 5. Therefore, it provides higher efficiency in handling 1-to-M traffic, for the following reasons. Firstly, each node simply needs to direct the 1-to-M traffic to the nearest master using any routing algorithm (we used a simple partially adaptive algorithm called the *odd-even* since it offers path diversity [27]). Hence, there is no need for extra circuitry or complicated algorithms to build the multicast tree path and to determine the forking points. Secondly, due to one forking point in routing path (the nearest master), packets will be replicated only at the destination routers. This will reduce power consumption by eliminating the need for duplicated traffic to travel through costly (power hungry and already loaded) intermediate wires and routers.

In order to direct the packets to the destinations, each multicast packet header must have multicast-address-bits (MAB). This header field is a bit vector where each bit represents a node, and it is set if the node is a multicasting group member. Since multicast traffic may be generated in a slave node, thus if the nearest master is part of the multicast group, it must transmit the flit via SWI first before draining the flit by the local PE. Lastly, although this approach is simple and efficient, 1-to-M routing is a major topic and many novel approaches can be further explored. However, it is out of this paper scope.

C. Multicast Challenges Facing The Proposed Architecture

The SWI layer can be represented logically as a multi-bus topology with multiple master nodes with Tx/Rx capability and multiple slave nodes with Rx capability. Each master has its own dedicated physical bus (frequency channels). However, each slave can receive from one master at a time, which creates competition between masters. This competition escalates as the average destinations of multicast flows and the number of master increase. As a result of this competition, two scenarios might develop: channel starvation and a multicast dependency deadlock. The first issue results when master(s) win the allocation of slaves repeatedly while other master nodes are

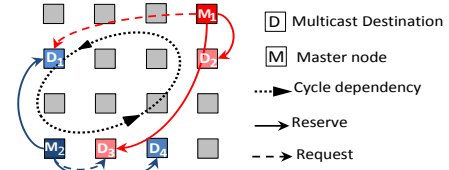


Fig. 6: Demonstration of the deadlock problem created by the multicast dependency.

waiting. The second scenario can be explained in an example as shown in Fig. 6. This figure demonstrates the deadlock problem resulting from multicast dependencies between two masters M_1 and M_2 requesting slaves $\{D_1, D_2, D_3\}$ and $\{D_1, D_3, D_4\}$, respectively. In this example, M_1 manages to allocate $\{D_2, D_3\}$ and now is waiting for D_1 , while M_2 allocates $\{D_1, D_4\}$ and now is waiting for D_3 . This scenario will cause a deadlock situation, since each master will not release its allocated slaves unless it delivers the rest of the packet flits.

For unicast traffic, the SWI utilization problem is master-centric where utilizing masters transmitter determines the performance. However, for multicast, this problem is shifted to be slave-centric where the utilization of the slaves receiver would determine the performance. Therefore, the use of virtual channels (VCs) for the SWI (VSWI) might offers the best utilization of SWI since it enables slaves to listen virtually to more than one master. This way, if a master is waiting for a message to be delivered to the rest of slaves or this message needs to be drained locally by the same master, idle reserved slaves will not be prevented from accepting traffic from another master. This complicates the allocation problem in finding a legal match between $masters \times VSWI \times slaves$ (three dimensions) so that no two masters are allocated the same VSWI for the same slaves simultaneously. Therefore, to offer fair deadlock-free arbitration while efficiently utilising the W-SWI, a set of solutions is proposed in the following sections.

D. Deadlock-free Centralized channel Allocation

This section presents the design of the global multiresources arbiter (GMA) and its rationale in addressing the contention problems mentioned above in a centralized approach. The resulting hybrid architecture with GMA will be referred to as W-SWI-C. To avoid the multicast cycle dependency scenario mentioned earlier, slaves could be allocated as a group. This can be achieved in the arbitration request masking stage by using the MAB-Check unit. This unit will not validate a request from any master unless all the slaves that have been requested are free by comparing the request with the content of a GMA reservation table. In this way, the arbitration problem is also minimised from the three-dimensional matching $masters \times VSWI \times slaves$ to the two-dimensional matching $masters \times VSWI$.

The main crucial feature in multi-resource allocations is legal matching where an output is assigned to one input and vice versa. Moreover, in order to minimise the decision latency, high arbitration parallelism is required. These two features can be achieved between the vector of inputs, which represents the masters, and a vector of outputs, which represents the

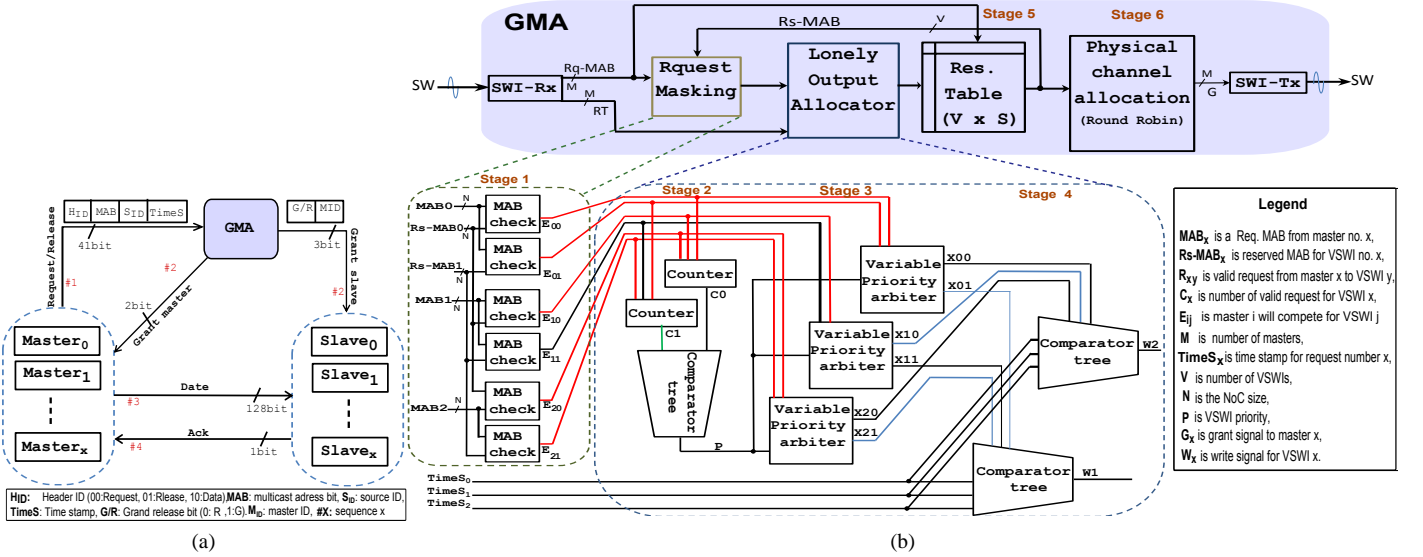


Fig. 7: (a) Demonstration control signals of SWI communication protocol exchanged between master, slaves and the GMA. (b) Design of the proposed global multi-resource arbiter (GMA) for SWI channels: stage (1) request masking; stages (2-4) achieve legal match with lonely output allocator [2]; stage (5) generates the grant signals for a fixed period. The figure also shows an example of GMA stages 1-4 with four masters and two VCs. Master (M_4) related logic is not drawn, for simplicity, but it is currently allocating some of the slaves requested by M_2 .

TABLE I: Notations used in this paper.

M_x	Master node number x	$P_t(S_x)$	Probability that S_x is free at time t
S_x	Slave node number x	$Rq_{x,y}$	Request from M_x to S_y
N	NoC size	MG_i	Set of multicast group members
N_m	Number of masters	Tag	ID-tagging for flow control
N_v	Number of VCs	ST	Time when the multicast flit is delivered to all its destinations
F	Flit	RV_x	Request vector from M_x
RT	Reservation table	$Gr_{x,y}$	Grant signal from S_y to M_x
RR	Round-robin arbiter	GMA	Global multiresources arbiter
T_x	Time slot sequence x	ZLL	Zero-load-latency
MAB	Multicast-address-bits	$VSWI$	VC of SWI
v_i	Input VC		
v_o	Output VC		

VSWI, by using two sets of arbitration: one for the input and one for the output. However, this is likely to lead to a poor legal matching or minimum matching, where less than optimal possible resources have been allocated. Optimum legal matching (maximum matching) can be achieved by adopting a lonely (or least-requested) output allocator (LOA) that introduces one more stage before input arbitration [2]. This extra stage counts the number of valid requests for each output in order to detect the level of competition over each output. Then, the less popular output will be given higher priority in the next arbitration stage. This should minimize conflict and produce maximum matching whenever possible.

Fig. 7b shows the structure of the proposed GMA which achieves the best legal match in two cycles (given that the requested resources are free) and with remarkably low circuit complexity. When the request is received from masters via SWI, they are demodulated and the request data are extracted, such as the requested destination(s) (MAB), a time-stamp and a source-ID. The first stage is a request masking in order to check if the master's request is possible for any of the VSWI by comparing its MAB with already reserved resources in the reservation table. The next three stages (2-4) represent the LOA which achieves the maximum matching of master-to-VSWI by minimising conflict between master requests over

VSWIs. This is accomplished in stage 2 by counting the valid requests for each VSWI, and then generating the priority signal that will prioritize a VSWI that is subject to less competition. Afterwards, in stage 3, each request will elect one of the VSWIs to compete over it. This elected VSWI should have less competition over it and the slaves requested are currently free. The final stage of LOA is stage 4, where the oldest request competing for each VSWI will be the winner out of comparator tree arbiters. The LOA is followed by stage 5, where the winning request from earlier stages will be stored in a reservation table. The size of this table is proportional to the NoC size, the number of master nodes (SWI channels), and the number of VSWIs.

The final stage 6 represents physical channel allocation. This stage alternates grant signal among subgroups of reserved slaves limited number of clock cycles. Each subgroup of slaves reserved under the same VSWI. This stage utilizes the VSWI to provide higher performance by allowing non-conflicting masters to transmit at the same time using their own channel frequencies. This stage consists mainly of a simple arbiter such as a round-robin arbiter (RR). The duration of the allocation can be tuned so that the allocation period is either (1) for one cycle, (2) for a fixed period, which would need a frequency divider, or (3) for as long as the request is asserted, which would need a hold release mechanism [2], in short, Hold. Tuning between these options mainly depends on traffic pattern and system-level evaluation, as discussed in Section IV-A. The final step is where the output is stored in an allocation register, and will be transmitted as a grant signal through an SWI-specific frequency control channel. The time cost of the arbitration in case of winning the arbitration is two clock cycles. Otherwise, the delay would equal to the arbitration cost plus a blocking period (T_b), which could be up to $Packet_{size} \times N_v$, where N_v is the number of VCSWs

if there is no congestion in the slaves.

To illustrate the GMA functionality, Fig. 7b also shows an example of the first 4 stages of the GMA that serves four masters with two VSWIs. However, the logic related to the forth master (M_4) is not shown, for simplicity, because it is currently merely reserving S_3 via $VSWI1$. Masters M_1, M_2 , and M_3 have requested the slaves $\{S_1, S_2\}$, $\{S_1, S_2\}$, and $\{S_2, S_3\}$ respectively. Since S_3 is already reserved via $VSWI1$, $E11$ is the only signal deactivated (not color red) by the MAB-check unit. In stage 2, therefore, the output priority (P) will be for the competition over the less-requested VSWI, which is $VSWI1$. Then, through the priority arbitration in stage 3, $M1$ and $M3$ will compete over $VSWI0$ while $M2$ will compete over $VSWI1$ alone, which are highlighted in blue. The winner of stage 4 ($W1$) will be the master request with the oldest time-stamp ($TimeS$).

The communication protocol among masters, slaves and the GMA taking place at the SWI level is shown in Fig. 7a. In order to utilize the limited available frequency bandwidth, the master interface sends a request on the same master data frequency channel (channel establishing phase). This request is identified by the header ID (H_{ID}) which distinguishes between request, release and data flits. In addition, the request packet consists of the data required for the arbitration process, such as the MAB, $TimeS$ and Source ID (S_{ID}). When the arbiter grants the request, it will generate two types of signals, which are the master and slave grant signals that grants their requests for the next cycle. These signals consists of two parts: the grant/release bit (G/R) and the requested master number (M_{ID}) that will inform the slave which channel it should listen to. After these signals are received, the data handshake phase starts. The master TxRx sends a data flit and waits for all the slaves in its multicast group acknowledge signals. Thus, the rest of the control signals require a bandwidth of 104 bits (for 4 masters and 24 slaves).

E. Efficient Decentralized Resource Allocation

This section proposes an alternative technique to handle multicast challenges in the proposed architecture, which is stretched multicast. This approach enables a master to transmit multicast flow to any number of currently free slaves and to retransmit it to the rest later. Although this means partial retransmission, it allows the concurrent execution of several overlapping multicast communications. Consequently, the decision should be determined at the slave end in a decentralized manner. This can be realized by any simple independent fair arbiter in each slave. Round-robin (RR) has been chosen since it provides stronger fairness than rotating or random arbiters and requires less circuit complexity than matrix and queuing arbiters [2]. There are many possible scenarios where this scheme will show better contention handling, fairness, and higher SWI utilization. For instance, Fig. 8 shows a comparison of contention handling of W-SWI-C with two VCSWs and a decentralized architecture (W-SWI-D). Clearly, even though we assume that the W-SWI-C manages to allocate all MG_i at T_1 , it offers less fairness and the multiplexing between flows is limited to the packet level.

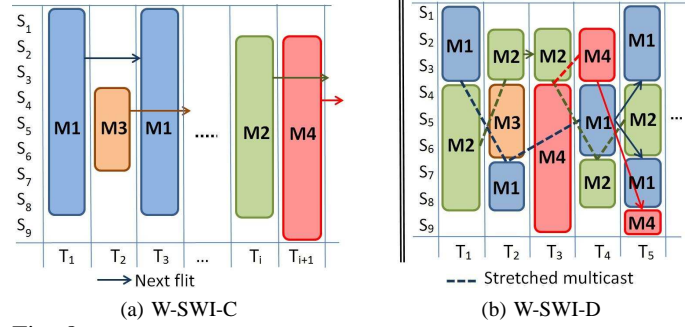


Fig. 8: Example shows traffic flows from four masters multiplexed in SWI where (a) W-SWI-C architecture (centralized arbitration), and (b) W-SWI-D architecture (decentralized arbitration and stretched multicast). where M is a master, S is a slave and T is a time slot.

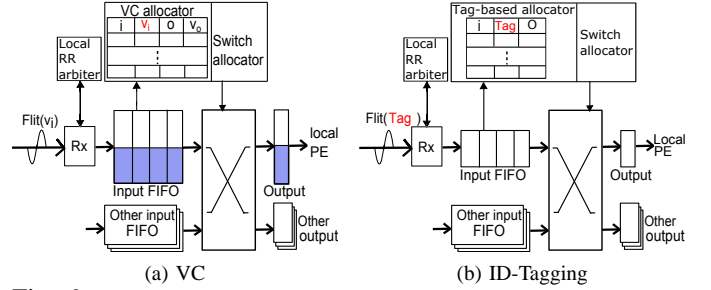


Fig. 9: Illustration of router micro-architecture with ID-Tagging based flow control and VC flow control.

However, since masters can allocate a subgroup of destinations, it might cause multicast dependency scenario, see Section III-C. In order to break multicast dependencies without the need to allocate all the requested slaves (MG_i) at a given time slot, each master should have its own virtual non-blocking channel for every slave. Thus, since each master already has its own physical channel frequency, virtually non-blocking channels can be achieved at the slave router by using a statically allocated VC where each master transmits via one VC ($N_v = N_M$). However, we developed a more efficient solution in terms of power and area overhead called ID-tagging-based flow-control (Tag). This technique simply consists of tagging each flit (F) with the transmitter master's ID (M_x) so that $Tag = M_x$, where $Tag, M_x \in \{1, \dots, N_M\}$. Then, at the reservation table (RT), the allocation entry is distinguished based on the input buffer (i), Tag and the input VC (V_i , if the design include VCSW). For simplicity the ID-tag remains unchanged in the router output port while being drained for the local processing element (PE). Fig. 9 shows an example of a router micro-architecture providing two virtual non-blocking channels by using either ID-tagging or permanently allocated VC s. Obviously, ID-tagging allows designers to choose a virtual-channelless design, which requires less area and power overheads. For this example, if buffer resizing are limited to the router port linked to SWI, then our calculations estimate a reduction in area of $\sim 2\%$, and in power of $\sim 5\%$ by using ID-tagging. Moreover, ID-tagging gives the freedom to use the VC for other purposes such as multiplexing flows from the same master ($VCSW$) in cases of congestion.

Algorithm 1: Master/slave interfaces procedure to communicate via SWI.

```

Procedure: Slave interface..... $j$  : local node ID
1: for  $i = 1 \rightarrow N_M$  do
2:    $R_i = Rx(RV_i[j])$ ;
3: end
4: forall the  $i \in \{1, \dots, N_M\}$  IF all  $Gr_{i,j} == 0$  do
5:    $i = RR(R_1, \dots, R_{N_M})$ ;
6:    $Tx(Gr_{i,j} = 1)$ ;
7: end
8: if  $Rx(Flit) == 1$  then
9:    $Tx(G_{i,j} = 0)$ ;
10: end



---


Procedure: Master interface..... $i$  : local node ID
1: forall the  $P \in \{N, E, S, W, L, SW\}$  do
2:   if  $Reserve(Input - port, Destination, MAB) == SW$  then
3:     forall the  $j \in \{0, \dots, N - 1\}$  do
4:        $RV_i[j] = RV_i[j] \vee MAB[j]$ ;
5:     end
6:   end
7: forall the  $j \in \{0, \dots, N\}$  IF any  $Rx(Gr[j]) == 1$  do
8:    $Tx(Flit = FIFO())$ ;
9:    $RV_i[j] = MAB[j] = 0$ ;
10:  forall the  $j \in \{0, \dots, N\}$  IF all  $MAB[j] == 0$  do
11:     $FIFO.eject$ ;
12:  end
13: end
14:  $Tx(RV_i)$ ;
15:

```

To prove that this scheme breaks the multicast dependencies on the SWI layer, assume that we have requests ($Rq_{i,j} \in RV_i$ for a M_i to any slave ($S_{j \in \{1, \dots, N\}}$). These requests should be granted within a finite time (T_f). This is to prove, that for all $Rq_{i,j} = 1$, then $Gr_{i,j} = 1$ within T_f , where $Gr_{i,j}$ is a grant signal. The probability that a slave local arbiter (RR) grants a master's request at the next time slot T_x is $P_{T_x}(S_j) = \frac{1}{N_M}$. In the worst case scenario, M_i has just been granted in T_{x-1} and all masters are requesting S_j at T_x . Thus, M_i has to wait until the RR arbiter of S_j grants all other masters. Assuming the average delay to serve each master request is T_D . Therefore, the maximum waiting period is $= N_M \times T_D$ and thus $P_{T_x + N_M \times T_D}(j) = 1$. Therefore, the serving time ST_i needed for a flit to be delivered to all slaves and ejected from the output buffer in M_i is:

$$ST_i = \max \left\{ \bigcup_{T_x=T_0 \rightarrow \max(T_x)} \{T_x \times (Rq_{i,0}, \dots, Rq_{i,N})\} \right\} \quad (1)$$

where the maximum period for delivering each flit per slave is $= T_x + N_M \times T_D$ for all $Rq_{i,j}$, and $j \in \{1, \dots, N\}$, and T_0 is the request insertion time. As a result, $1 \leq ST_i \leq N_M \times T_D$. For instance, assume that M_1 send requests $\{Rq_{1,1}, Rq_{1,2}, Rq_{1,3}\}$ at T_0 . If each request has been granted in $\{T_3, T_3, T_1\}$ respectively then, according to Equation. 1, $ST_1 = T_3$. In other words, the maximum serving time for the whole multicast is the maximum serving time among all the multicast group members, which is a finite time.

The communication protocol for the decentralized approach has fewer control signal types than that in the centralized approach, but with slight composite interface procedure. Algorithm 1 shows the procedure used for master and slave interfaces. A master (M_i) interface sends a RV_i via SWI that consists of a request ($Rq_{i,j}$) to each slave (S_j) wherever

TABLE II: Comparison of highlighted features of the centralized and decentralized approaches for the proposed architecture.

Features	Centralized approach (W-SWI-C)	Decentralized approach (W-SWI-D)
Flit retransmission	none	up to $N_M - 1$
Probability of transmission	$P_t(S_x) * P_t(S_y) \dots * P_t(S_z)$	$P_t(S_j)$
Flow control	wormhole	wormhole on packet level and packet-switching on the flit level
VSWI utilization	on the slave end	on both master and slave end
Channel establishment	per packet	per flit

$MAB(j) = 1$; see lines 1 to 7. At the slave end, the local RR arbiter will determine which master it will listen to in the next time slot and send the $Gr_{i,j}$ signal to, see lines 4 to 7. Both $Rq_{i,j}$ and $Gr_{i,j}$ are transmitted via SWI using On-off keying (OOK) modelling for simplicity. When the RR grants the request, the data handshake phase starts by sending the data flit to all slaves who responded to the request and waits for them to acknowledge reception before resetting the requests intended for them. The adopted handshaking protocol is a non-return-to-zero protocol [28]. The master interface keeps track of which slave has been served by updating the MAB of the current flit, as shown in line 10. Then, if all MG_i members have received the multicast flit, it will be ejected and a new set of requests might be sent based on the next flit MAB .

F. Theoretical analysis and Comparisons

The previous two sections presented the techniques to address SWI layer multicast challenges. In this section, these techniques are analytically compared and discussed to serve as ground for understanding the evaluation results in the following sections.

In the centralized approach, a master has to wait until all the requested slaves (multicast group) are free in order to avoid deadlock. To mathematically express the problem in this approach, assume a master (M_i) is sending a request vector (RV_i) to the GMA to allocate a set of multicast-group (MG_i) $\subset \{S_0, \dots, S_{N-1}\}$, where S_j is a slave, and N is the NoC size). The probability that a slave ($S_j \in MG_i$) is free in time slot t is denoted by $P_t(S_j)$. Therefore, the probability that the M_i request ($Rq_{i,j}$) will be granted is the intersection probability ($P_t(S_x) * P_t(S_y) \dots * P_t(S_z)$). This is clearly less or equal to $P_t(S_j)$ for all $S_j \in MG_i$, which is the case of W-SWI-D. This will keep free requested slaves idle until all the members of MG_i are free.

In terms of blocking period, although the same flit might be retransmitted (up to N_M , where N_M is the number of master nodes), the stretched multicast offers higher fairness and utilization of the SWI. This is because the reallocation of slaves on flit bases prevents flows from blocking each other. In contrast, the blocking period ($Time_{block}$) in W-SWI-C can be predicted to be:

$$Time_{block} = (P_{size} \times N_v \times Slice) + Time_{congestion} \quad (2)$$

where N_v is the number of VC, $Slice$ is time slot per flit, $Time_{congestion}$ is the duration of any congestion in the master (the channel owner) or the slave that would cause idle time slots. However, $Time_{congestion}$ might equal zero if there is

TABLE III: The parameters adopted for the NoC-based multi/many-core tile. This system tile size is $3.6 \times 5.2 \text{ mm}^2$.

IP components		NoC components	
Two Pentium™ class IA-32 cores	Message passing router	4-port to neighbour routers, 1-port to local cores and 1-port for the SWI.	6 buffers each with 4-flit(128 bit) depth and 4 virtual channels
Two 256 KB private L2 caches	Links	5 bidirectional wire interconnects (16 byte width).	1 surface wave channel (Rx or TxRx), 32 sub-channel with 16-QAM modulation.

no heavy load at either ends. Thus, the stretched multicast improves fairness by reducing the blocking period between flows. However, the overhead of channel establishment per flit in decentralized approach might overcome the SWI utilization improvements. Table II summarize the comparison of the main features of the centralized and decentralized approaches. These features will rationalize some of the results in the next section.

IV. SYSTEM LEVEL EVALUATION AND DISCUSSION

This section presents results obtained from our cycle-accurate NoC simulator which was built by modifying the existing Noxim simulator [29] for the W-SWI-C, W-SWI-D, virtual circuit tree multicast (VCTM), and the baseline architecture (MeshS). The MeshS in this paper refers to a wire-based regular mesh NoC that manages the 1-to-M traffic as a software multicast. In this paper, the Intel single-chip cloud computer (SCC) [30] is adopted as the baseline architecture. This chip is designed for performance critical many-cores, which makes it optimal for the purpose of this study. Table III shows the modified tile specifications. Packet sizes of 1, 4 and 12 flits were chosen as an example to demonstrate the behaviour of the proposed architecture under packet-switching, virtual-cut-through-switching and wormhole-switching flow control respectively. The number of master nodes is based on the available frequency range for 45nm , which was estimated to be four channels (plus the frequencies specified for control signals). However, this frequency range is scaling with technology [8]. As a result, the number of master nodes was increased when simulating larger NoCs, assuming that the technology will have been scaled too. In addition, a VSWI number of four was chosen, which realizes a better performance/cost trade-off. In addition, in the evaluation in this section the VC was chosen to be equal to the VSWI for simplicity of router architecture.

The simulation was conducted with synthetic traffics, which are: (1) Random, where packets are transmitted randomly with uniform probability to other nodes; (2) Hotspot, which is the same as the random but with specific nodes called hotspots, four in this case, with higher probability of traffic dispatched to them; (3) Transpose, where a node sends a packet to other node that has its address transposed. These synthetic traffic adjusted to inject a specific percentage of broadcast (1-to-all) or multicast (1-to-M). The source nodes of this multicast traffic are selected randomly during the simulation, while the rest of the traffic consist of normal unicast packets according to the named synthetic traffic. In addition, in the case of multicast, the destinations of these packets are also selected randomly. In addition, the evaluation of the proposed architecture and baseline architecture includes real application benchmarks whose details are shown in Section IV-D.

A. Performance Improvements

This section presents a performance evaluation of the proposed architecture under synthetic traffic. Fig. 10a shows much less average delay for the W-SWI-C over the MeshS with Random traffic consisting of 10% broadcast and 90% unicast. Even for zero-load-latency (ZLL), the average delay improvement is $\sim 22\text{x}$. Similar improvements are obtained with Transpose ($\sim 24\text{x}$) and Hotspot traffic ($\sim 21.8\text{x}$). Obviously, further improvements can be reported as the PIR increased since MeshS is saturated before W-SWI-C, as shown in Fig. 10a. These significant improvements are due to the software multicast in MeshS that replicates the multicast traffic to all destinations. Thus, it will increase the load and hotspots on the NoC. In contrast, In W-SWI, packets are replicated at the multicast destination routers and through a short-cut links that avoid costly intermediate routers and wire links. In addition, when multicast traffic is used, the resulting improvements were up to 12x over MeshS. This is due to the relatively lower load and hotspots caused by multicast compared to broadcast.

On the other hand, Fig. 10b shows that W-SWI-C with a Hold allocation is better than fixed period allocation with 10% multicast, see Section III-D. This is due to the fact that the hold-release mechanism eliminates the reallocation delay. Fig. 10b also shows that, as the allocation period is increased, performance starts to decay because of the inflexibility of resources time scheduling. Fig. 10c shows a comparison of W-SWI-C and W-SWI-D for different values of N_v . Clearly, the proposed W-SWI-D shows slightly better performance than W-SWI-C with one VC (with an improvement of $\sim 5\%$). This is due to the fact that with one VC, ID-tagging allows each master to have a virtually non-blocking channel. Moreover, the performance results show that even for higher N_v , the W-SWI-D is better before reaching $1.5 \times \text{ZLL}$. However, after this point the multiplexing delay between masters starts to overcome the improvements in the SWI utilization. Therefore, the W-SWI-C curves show more linear performance against the increase in the PIR. Fig. 10d demonstrates the effect of increasing the number of masters in W-SWI-C: Hold under 10% multicast. Clearly, the performance in general improves as the number of SWIs is increased. However, the W-SWI-C with two masters and two physical channels (SWI:2) seems to slightly outperform the W-SWI-C:3. This could be because the increase in SWI channels will increase contention on the shared medium and the arbitration delay might impact on the performance. Thus, the optimum number of masters is not always the highest. In this work, the SWI:4 is chosen as the design parameter since it offers the best performance/cost trade-off, in addition to the fact that it copes with the frequency limit for the 45nm technology.

B. Power Reduction

This section presents the evaluation results power consumption. The router's static and dynamic power is calculated using Orion 2.0 [31] area and power models including the extra RR for SWI in the case of W-SWI-D. The modelled router power has been calibrated to match the reported power measurement of the implemented NoC [30]. In addition, Power

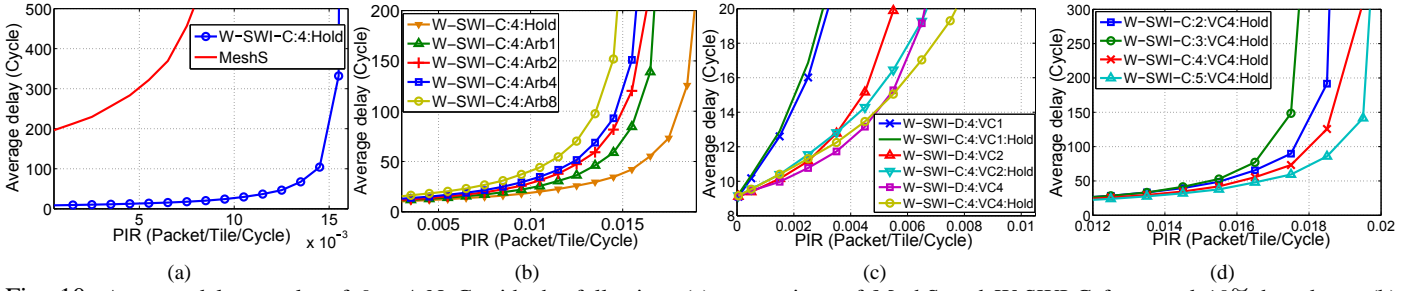


Fig. 10: Average delay results of 6×4 NoC with the following: (a) comparison of MeshS and W-SWI-C for a and 10% broadcast; (b) W-SWI-C with different allocation techniques; (c) comparison of W-SWI-C and W-SWI-D with different VC numbers; (d) W-SWI-C with different SWI master number. Note that W-SWI- $N_m:VCN_v:ArbP$ refers to W-SWI with N_m number of masters, N_v number of VC and P number of grant cycles. Hold is where GMA grants a master until all its current data flow is transmitted.

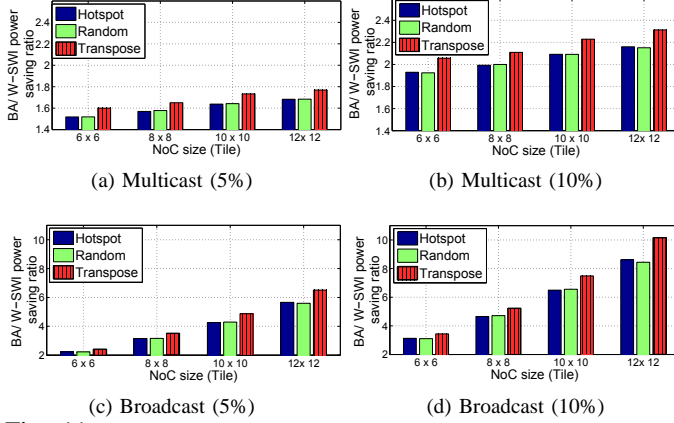


Fig. 11: Communication power saving ratio of the W-SWI over the MeshS for different network sizes, types of synthetic traffic and broadcast/multicast ratios.

dissipation for wire links is calculated for the horizontal links (3.6mm) and the vertical links (5.2mm), according to the SCC measurements [30]. The transceiver (TxRx) power consumption projection [8], [22] is used, which is calculated to be 24mW per sub-channel. The SWI power dissipation is also calculated based on the analytical model introduced previously [23]. The GMA was designed using Verilog and then synthesized using the Synopsys Design Compiler and mapped onto the PDK 45nm technology library to calculate its dynamic power (4.8mW) and leakage power (59.3μW). The PDK 45nm library selected to match the 45nm technology of the baseline architecture and due to its availability when this study had conducted. Then all these values were used in adjusted Noxim to calculate the overall NoC power consumption.

Fig. 11 shows the ratio of the MeshS power consumption over the power consumption of the W-SWI-C at $PIR = 2 \times ZLL$ for different NoC sizes, synthetic traffics and percentages of 1-to-M. Significant improvements in the NoC power consumption reduction ratio are demonstrated. For instance, the power ratio of MeshS to W-SWI starts from more than double ($\sim 2x$) and increases up to $\sim 10x$ as the NoC size and the broadcast percentages are increased. However, less improvement appears in the case of multicast. This is because the multicast group members are fewer, which reduces

TABLE IV: Average delay and PIR at the edge of NoC saturation comparison of the W-SWI and VCTM.

Packet size (flit)	Multicast (%)	Average delay (Cycle)		PIR	Improvements (VCTM/W-SWI)
		W-SWI	VCTM		
1	5	16.74	29.65	0.315	1.77
	10	16.68	28.87	0.18	1.73
	15	16.61	29.07	0.125	1.75
4	5	16.86	96.71	0.065	5.73
	10	16.33	88.75	0.04	5.43
	15	16.6	90.5	0.03	5.45

the utilization of the SWI fan-out feature. Nonetheless, it still shows remarkable improvements, increasing from $\sim 1.5x$ to $\sim 2.3x$ proportionally to NoC size and 1-to-M ratio. On the other hand, the W-SWI-D achieved generally lower improvements than W-SWI-C due to retransmission and the arbitration overhead. However, the W-SWI-D might outperform the W-SWI-C in case of a low load ($\sim 1.5 \times ZLL$) due to the higher SWI utilization, as shown in our previous work [16]. In general, these new findings prove that the W-SWI has a remarkable scalability and effectiveness in mitigating 1-to-M communication issues.

C. Comparison with Related Work

In this study, one of the state-of-the-art wire-based NoCs with a tree-based multicast scheme was replicated in order to compare it with the proposed architecture. This scheme is the VCTM [3]. It has been chosen because of its efficiency and simplicity, where a minimum of modifications to the baseline router architecture are required since the VCTM basically enables a mesh NoC to have a tree-based routing capability. These modifications mainly include the VCTM table and the control circuits for the forking of one flit/cycle. These features will provide a fair comparison with the proposed architecture. This scheme is based on assigning one of the VCTM table entries in each router to every new multicast group with a unique source. Then, packet forking and routing is conducted according to the VCTM table. This look-up-table needs a set-up stage to define its content for each new multicast group introduced to the NoC. The set-up stage use software multicast and the table entry is cumulatively set up. Thus, the authors acknowledged that interconnect performance is based on the ratio of the VCTM table size in the router to the number of unique multicast groups injected to the NoC [3].

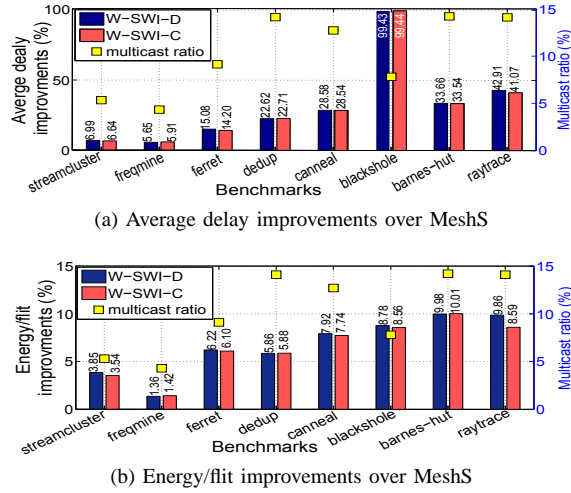


Fig. 12: Comparison between the average delay and energy improvements of W-SWI-C and W-SWI-D over MeshS under real applications benchmarks from PARSEC [32] and SPLASH2 [33] for 10×8 NoC.

A big limitation of the VCTM is the inability to handle wormhole traffic due to deadlock occurrence. Therefore, the chosen packet sizes were one and four flits to give a fair demonstration of the NoC's performance compression under packet-switching and virtual-cut-through-switching respectively. In addition, the VCTM is limited to a turn-model routing with no path diversity. Otherwise, deadlock problems could appear because diversity might introduce cycles when building the multicast tree in the set-up stage. In contrast, path diversity is a favourable feature in our architecture and has been tackled by using odd-even routing, see Section III-B. Therefore, XY routing was used in all of the simulations in this section in order to provide a fair comparison. Moreover, according to Jerger *et al.* [3], a VCTM with 512 entries per source offers good performance/cost levels in most cases. Therefore, a VCTM with 512 entries was considered in all our evaluations. Table IV shows a performance comparison of the W-SWI-C and the VCTM architectures with different multicast ratios. The multicast source and group members were selected randomly. The average delay and PIR is reported for the NoC saturation edge where the average delay is double the ZLL. In this PIR, the proposed architecture shows steady improvements of around $\sim 1.8X$ and $\sim 5.5X$ for one and four flit packet sizes respectively. Although the proposed architecture performs better with wormhole or virtual-cut-through switching, its performance is still almost double that of the VCTM under basic packet switching.

D. Evaluation based on Real Application Benchmark

In order to demonstrate the effectiveness of the proposed architecture for real applications, a set of application benchmarks from a standard suits, which are PARSEC [32] and SPLASH2 [33], have been considered. These benchmarks are built based on the traffic analysis of communication trace-files generated from the many-core simulator [34] where all the benchmark applications were run with MESI cache coherence protocol. This protocol is well-known and used in many multi-processor systems [35]. As a result, based on this traffic

TABLE V: Area overhead evaluation for W-SWI-C, proposed W-SWI-D and VCTM-512 [3] over baseline architecture (MeshS).

NoC component		Area per item (mm^2) for 45nm technology				
Component	No.	MeshS	W-SWI-C	W-SWI-D	VCTM	RF-I
Router	24	1.0853	1.5124	1.5931	1.0853	1.5124
Transmitter	4	-	0.1558	0.1558	-	0.1558
Receiver	24	-	0.0083	0.0083	-	0.0083
Global arbiter	1	-	0.0552	-	-	0.0552
Local arbiter	24	-	-	0.0309	-	-
VCTM table	24	-	-	-	2.3608	-
Wire Links	1	13.653	13.653	13.653	13.653	13.653
Total extra area over MeshS= (all components \times their No.)- MeshS area (mm^2)			11.13	13.75	56.66	11.9
NoC/SCC-die area (%)		7	8.96	9.29	16.99	9.1

analysis a synthetic traffics have been built, which have the same injection rates, packet size and source/destination(s) of each multicast and unicast traffic flows in these application benchmarks. These synthetic traffics then run for a million cycles with our cycle-accurate system-level NoC simulator.

Fig. 12a presents the performance improvement gained using the proposed W-SWI-C and W-SWI-D architectures over the MeshS for a NoC size of 10×8 . In general, the average delay improvements of the W-SWI-C and W-SWI-D over MeshS are almost similar and range from ~ 5 to $\sim 99\%$. Moreover, these improvements are clearly proportional to the percentage of the multicast's ratio from the total PIR (4 to 14.2%). An exception is the case of the blackhole benchmark, where the improvement over MeshS is around 99%, even though the multicast ratio is 7.8%. This is due to the nature of the traffic hotspots (specifically multicast source hotspots) that cause the traffic source to quickly become saturated. In contrast, the proposed architecture's have more ability to effectively alleviates such traffic source overload and therefore reduce the serialization delay of multicast traffic into separated unicast traffic. On the other hand, Fig. 12b also demonstrates the average energy/flit improvements over MeshS, where a flit is 128 b as mentioned earlier. Once again, W-SWI-C and W-SWI-D achieved better rates of energy/flit over MeshS of up to $\sim 10\%$ and the improvements are proportional to the multicast percentage. However, most of these benchmarks run with relatively low PIR. Therefore, as shown in Section IV-A, the W-SWI-D outperform the W-SWI-C under low load. As a result, even though the W-SWI-D uses retransmissions that increase power consumption, it is better than W-SWI-C in some benchmarks in terms of power. In general, these results prove the potential of the proposed architectures for future NoC-based many-cores.

E. Area Overhead Evaluation

It is essential to evaluate chip area overheads for the extra on-chip circuits required for the proposed architecture. Firstly, it is assumed that the active area calculated for transceivers in previous research [8] is the only part scaled down when moving to 45nm technology, while the passive parts remain almost the same since they are proportional to the channels' operational frequency range. Therefore, the projected transmitter area is $4870\mu m^2$ per sub-channel, while the projected receiver area is $260\mu m^2$ per sub-channel, where the active area is proportional to the square of the scaling factor [36].

Secondly, the area of baseline router and the extra router port (buffer, crossbar and related circuits) is calculated using the Orion 2.0 [31] model as $0.427mm^2$. The modelled baseline router area is a 6% less than the reported implemented router area [30], which is acceptable for the purpose of comparison evaluation in this paper. Thirdly, the GMA (for W-SWI-C) and RR (for W-SWI-D) was designed using Verilog and then synthesized using the Synopsys design compiler and mapped onto the PDK $45nm$ technology library to calculate its area. Their area was found to be $0.0114mm^2$ and $0.0002mm^2$, respectively, and to which the TxRx (for control signals) estimated area of $0.0438mm^2$ and $0.0307mm^2$, respectively, was added. Likewise, the VCTM with a 512 entry/source lookup table was designed using Verilog and then synthesised. Moreover, to compare other emerging interconnects, the RF-I's transmission line area was calculated and considered to be routed through the chip (NoC size 6×4) as a U shape passing through all nodes [8]. A transmission line with $12\mu m$ pitch has been considered in calculations of RF-I area overhead.

Table V shows area overhead breakdown for the MeshS, proposed W-SWI-C, W-SWI-D, VCTM and RF-I. RF-I, has been chosen as an example to compare with emerging interconnects, full comparison with other interconnects is shown previous study [21]. Obviously, most of the extra area overhead for the W-SWI-C and W-SWI-D, of around 1.9% is due to the extra router port. However, the W-SWI-D area overhead is higher than the W-SWI-C ($\sim 2.4\%$, $\sim 2\%$, respectively). This is mostly due to increasing the VC allocation unit area in all routers to implement the ID-tagging scheme. Moreover, the W-SWI-C offers a better die area-performance trade-off compared to RF-I transmission lines that offer the same connectivity [8], since fat transmission lines need to be implemented through the chip. Not only that, but the W-SWI-C also beats the VCTM in area overhead (around 5 times less). Therefore, the W-SWI-C succeeds these architectures in terms of low area overheads.

V. CONCLUSION AND FUTURE WORK

This paper tackle the 1-to-M traffic issues efficiently using the hybrid wire-SWI architecture for on-chip communication. Zenneck surface wave low power dissipation, high signal propagation speed and fan-out capability all contributes to significantly mitigate the 1-to-M communication issues that the NoC-based many-core processors in particular suffers from. In addition, novel, efficient, and deadlock-free centralized (W-SWI-C) and decentralized (W-SWI-D) arbitration and allocation techniques along with a multicast routing scheme for this architecture are proposed and discussed. The evaluation results show significant improvements in terms of average delay, saturated PIR and power consumption with a relatively small die area penalty compared to state-of-the-art-architectures. Moreover, the comparison of the W-SWI-C and the W-SWI-D has proven that the former is preferred for higher traffic loads while the latter is optimal for low traffic loads. In general, the results demonstrate the high scalability of the W-SWI for the many-cores era. Future work should include the investigation of many-to-one traffic patterns.

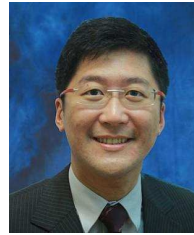
REFERENCES

- [1] S. Borkar, "Thousand core chips: a technology perspective," in *Proceedings of the 44th annual Design Automation Conference, DAC '07*, (New York, NY, USA), ACM, 2007.
- [2] W. Dally and B. Towles, *Principles and Practices of Interconnection Networks*. Morgan Kaufmann, 2004.
- [3] N. Jerger, L.-S. Peh, and M. Lipasti, "Virtual circuit tree multicasting: A case for on-chip hardware multicast support," in *ISCA '08. 35th International Symposium on Computer Architecture*, pp. 229–240, 2008.
- [4] R. Manevich, I. Walter, I. Cidon, and A. Kolodny, "Best of both worlds: A bus enhanced noc (benoc)," in *Networks-on-Chip, 2009. NoCS 2009. 3rd ACM/IEEE International Symposium on*, pp. 173–182, May 2009.
- [5] Semiconductor Industry Association, "ITRS: International Technology Roadmap for Semiconductors ." <http://www.itrs.net/reports.html> [online], 2011.
- [6] R. Ho, K. Mai, and M. Horowitz, "The future of wires," *Proceedings of the IEEE*, vol. 89, pp. 490–504, apr 2001.
- [7] A. Ganguly, K. Chang, S. Deb, P. Pande, B. Belzer, and C. Teuscher, "Scalable hybrid wireless network-on-chip architectures for multicore systems," *IEEE Transactions on Computer*, vol. 60, no. 10, pp. 1485–1502, 2011.
- [8] M. C. F. Chang, J. Cong, A. Kaplan, C. Liu, M. Naik, J. Premkumar, G. Reinman, E. Socher, and S.-W. Tam, "Power reduction of cmp communication networks via rf-interconnects," in *Microarchitecture, 2008. MICRO-41. 2008 41st IEEE/ACM International Symposium on*, pp. 376–387, Nov 2008.
- [9] A. Carpenter, J. Hu, J. Xu, M. Huang, H. Wu, and P. Liu, "Using transmission lines for global on-chip communication," *Emerging and Selected Topics in Circuits and Systems, IEEE Journal on*, vol. 2, pp. 183–193, June 2012.
- [10] S. Deb, A. Ganguly, P. Pande, B. Belzer, and D. Heo, "Wireless noc as interconnection backbone for multicore chips: Promises and challenges," *Emerging and Selected Topics in Circuits and Systems, IEEE Journal on*, vol. 2, pp. 228–239, June 2012.
- [11] P. Dong, Y.-K. Chen, T. Gu, L. L. Buhl, D. T. Neilson, and J. H. Sinsky, "Reconfigurable 100 gb/s silicon photonic network-on-chip [invited]," *Optical Communications and Networking, IEEE/OSA Journal of*, vol. 7, pp. A37–A43, Jan 2015.
- [12] D. Miller, "Device requirements for optical interconnects to silicon chips," *Proceedings of the IEEE*, vol. 97, pp. 1166–1185, july 2009.
- [13] J. Turner, M. Jessup, and K.-F. Tong, "A novel technique enabling the realisation of 60 GHz body area networks," in *Wearable and Implantable Body Sensor Networks (BSN), 2012 Ninth International Conference on*, pp. 58–62, may 2012.
- [14] J. Hendry, "Isolation of the zenneck surface wave," in *Antennas and Propagation Conference (LAPC), 2010 Loughborough*, pp. 613–616, nov. 2010.
- [15] A. Karkar, N. Dahir, R. Al-Dujaily, K. Tong, T. Mak, and A. Yakovlev, "Hybrid wire-surface wave architecture for one-to-many communication in networks-on-chip," in *Design, Automation and Test in Europe Conference and Exhibition (DATE), 2014*, pp. 1–4, March 2014.
- [16] A. Karkar, K. Tong, T. Mak, and A. Yakovlev, "Mixed wire and surface-wave communication fabrics for decentralized on-chip multicasting," in *Proceedings of the 2015 Design, Automation & Test in Europe Conference & Exhibition, DATE '15*, pp. 794–799, EDA Consortium.
- [17] R. Morris, E. Jolley, and A. Karanth Kodi, "Extending the performance and energy-efficiency of shared memory multicores with nanophotonic technology," 2013.
- [18] H.-M. Hsu, T.-H. Lee, and C.-J. Hsu, "Millimeter-wave transmission line in 90-nm cmos technology," *Emerging and Selected Topics in Circuits and Systems, IEEE Journal on*, vol. 2, pp. 194–199, June 2012.
- [19] D. DiTomaso, A. Kodi, D. Matolak, S. Kaya, S. Laha, and W. Rayess, "A-winoc: Adaptive wireless network-on-chip architecture for chip multiprocessors," *Parallel and Distributed Systems, IEEE Transactions on*, vol. PP, no. 99, pp. 1–1, 2014.

- [20] A. Mineo, M. Palesi, G. Ascia, and V. Catania, "An adaptive transmitting power technique for energy efficient mm-wave wireless nocs," in *Design, Automation and Test in Europe Conference and Exhibition (DATE), 2014*, pp. 1–6, March 2014.
- [21] A. Karkar, T. Mak, K. F. Tong, and A. Yakovlev, "A survey of emerging interconnects for on-chip efficient multicast and broadcast in many-cores," *IEEE Circuits and Systems Magazine*, vol. 16, no. 1, pp. 58–72, 2016.
- [22] M.-C. Chang, V. Roychowdhury, L. Zhang, H. Shin, and Y. Qian, "Rf/wireless interconnect for inter- and intra-chip communications," *Proceedings of the IEEE*, vol. 89, pp. 456–466, Apr 2001.
- [23] A. Karkar, J. Turner, K. Tong, R. Al-Dujaily, T. Mak, A. Yakovlev, and F. Xia, "Hybrid wire-surface wave interconnects for next-generation networks-on-chip," *Computers Digital Techniques, IET*, vol. 7, pp. 294–303, November 2013.
- [24] D. M. Pozar, *Microwave Engineering*. John Wiley & Sons, 2009.
- [25] Y. Wang, Y.-H. Han, L. Zhang, B.-Z. Fu, C. Liu, H.-W. Li, and X. Li, "Economizing tsv resources in 3-d network-on-chip design," *Very Large Scale Integration (VLSI) Systems, IEEE Transactions on*, vol. 23, pp. 493–506, March 2015.
- [26] A. Cilardo and E. Fusella, "Design automation for application-specific on-chip interconnects: A survey," *Integration, the {VLSI} Journal*, vol. 52, pp. 102 – 121, 2016.
- [27] G.-M. Chiu, "The odd-even turn model for adaptive routing," *Parallel and Distributed Systems, IEEE Transactions on*, vol. 11, pp. 729 –738, jul 2000.
- [28] D. Kinniment, *Synchronization, Arbitration and Choice*. England: Wiley Publishing, 2007.
- [29] F. Fazzino, M. Palesi, and D. Patti, "Noxim: Network-on-chip simulator," 2010.
- [30] P. Salihundam, S. Jain, T. Jacob, S. Kumar, V. Erraguntla, Y. Hoskote, S. Vangal, G. Ruhl, and N. Borkar, "A 2 tb/s 6×4 mesh network for a single-chip cloud computer with dvfs in 45 nm cmos," *Solid-State Circuits, IEEE Journal of*, vol. 46, pp. 757–766, April 2011.
- [31] A. Kahng, B. Li, L.-S. Peh, and K. Samadi, "Orion 2.0: A power-area simulator for interconnection networks," *Very Large Scale Integration (VLSI) Systems, IEEE Transactions on*, vol. 20, pp. 191–196, Jan 2012.
- [32] C. Bienia, S. Kumar, J. P. Singh, and K. Li, "The parsec benchmark suite: Characterization and architectural implications," in *Proceedings of the 17th International Conference on Parallel Architectures and Compilation Techniques, PACT '08*, (New York, NY, USA), pp. 72–81, ACM, 2008.
- [33] J. P. Singh, W.-D. Weber, and A. Gupta, "Splash: Stanford parallel applications for shared-memory," *SIGARCH Comput. Archit. News*, vol. 20, pp. 5–44, Mar. 1992.
- [34] J. Xue, A. Garg, B. Ciftcioglu, J. Hu, S. Wang, I. Savidis, M. Jain, R. Berman, P. Liu, M. Huang, H. Wu, E. Friedman, G. Wicks, and D. Moore, "An intra-chip free-space optical interconnect," *SIGARCH Comput. Archit. News*, vol. 38, pp. 94–105, June 2010.
- [35] T. Shanley, *Pentium Pro Processor System Architecture*. Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc., 1st ed., 1996.
- [36] J. Srinivasan, S. Adve, P. Bose, and J. Rivers, "The impact of technology scaling on lifetime reliability," in *Dependable Systems and Networks, 2004 International Conference on*, pp. 177 – 186, june-1 july 2004.



Ammar Karkar Std. MIEEE, Std. MIET, received both the BSc degrees in computer engineering from the Almustanseria University in Baghdad/Iraq in 2004 and Msc degree (Hons) in Information computer and Network security from NYIT in Amman/Jordan in 2007. He is currently working toward the PhD degree with the School of Electrical and Electronic Engineering, Newcastle University/UK. He has an interest in exploring cutting-edge computing systems including networks-on-chip, emerging interconnects, many-cores, and VLSI design.



systems and emerging applications using novel architectures, algorithms. His newly proposed approaches using runtime optimization and adaptation strategies led to multiple prestigious Best-Paper-Awards from three major conferences, DATE'11, VLSI-SoC'14 and PDP'15.



Nizar Dahir received the BSc and MSc degrees in Computer Engineering from AL-Nahrain University, Baghdad, Iraq, in 1997 and 2000, respectively and the PhD degree in Computer Engineering from the School of Electrical and Electronic Engineering, Newcastle University, UK. He is now working as a research associate with the Intelligent Systems Group, Department of Electronics, University of York. His research interests are many-core systems and networks-on-chip.



Raed Al-Dujaily received both the BSc and MSc degrees in control and computer engineering from the University of Technology of Baghdad/Iraq in 1998 and 2001, respectively, and PhD degree from the School of Electrical, Electronic and Computer Engineering, Newcastle University, United Kingdom in 2012. His research interests include Network-on-Chip, reconfigurable computing, VLSI circuits design and fault tolerance in VLSI. He is a student member of the IEEE.



Kin-Fai Tong received the BEng(Hons) and PhD degrees in Electronic Engineering from the City University of Hong Kong. He worked an Expert researcher in the National Institute of Information and Communications Technology (NiCT), Japan, where his main research focused on photonic-integrated millimetre-wave planar antennas for Gbits wireless communication systems. Dr Tong is now a senior lecturer at the Department of Electronic and Electrical Engineering, University College London (UCL). Early in 1994, he has been credited to be one of the

first who introduced the idea of integrating microstrip patch antennas into mobile phones. Moreover, he pioneered in developing Finite Difference Time Domain (FDTD) models for the investigations of the ultra-wideband behaviour of U-slot microstrip patch antennas. He has co-authored two book chapters on planar antenna designs and is author or co-author of over 90 publications.



Professor Alex Yakovlev DSc, FIET, SMIEEE (AY, UoN) founded and leads the MicroSystems Research Group, and co-founded the Asynchronous Systems Laboratory at Newcastle University. He was awarded an EPSRC Dream Fellowship in 2011-13. He has published 8 edited and co-authored monographs and more than 300 papers in academic journals and conferences, most of which are in the area of concurrent and asynchronous systems. He has chaired program committees of several international conferences in this area, including the IEEE Int. Symposium on Asynchronous Circuits and Systems (ASYNC), Petri nets (ICATPN), Applications of Concurrency to Systems Design (ACSD), and he has been Chairman of the Steering committee of the Conference on Application of Concurrency to System Design since 2001. He has been principal investigator on more than 25 research grants and supervised 40 PhD students.